IN THE CLAIMS

Kindly amend claims 1, 4-6, 8, 9, 16, 24, 25, 27-31 and 33-36 as set forth below. In addition, please add new claims 37-42 and delete claims 21 and 32 without prejudice to, or disclaimer of, the subject matter disclosed therein. The subject matter of claims 21 and 32 has been placed into claims 16 and 27, respectively.

The following is a complete listing of revised claims with a status identifier in parenthesis.

LISTING OF CLAIMS

1.   (Currently Amended)   A caching method for building a table having n entries to select r most frequently used Internet site names from m Internet site names that can be received at a resource delivery site, so that the resource delivery site can cache the corresponding resources from the r most frequently used Internet sites in advance where $r \leq n$, the method comprising the steps of:

(a)   receiving an Internet site name from a packet;

(b)   storing the Internet site name in an entry of [[the]] a table having n entries if the Internet site name is not in the table; [[and]]

(c)   counting the number of times the Internet site name has been received, characterized in that $m > n$ and if the Internet site name is new and the table is full, an entry of the table is replaced selecting an entry from a set of replaceable entries in the table;

(d)   replacing the selected entry with the new entry; and

(e)   caching a resource corresponding to at least one of a most frequently used Internet sites *r* where *r* ≤ *n*.

2.   (Original)   The method of claim 1 wherein the Internet site name is a URL (Uniform Resource Locator).

3.   (Original)   The method of claim 1 wherein each entry of the table has a name field for storing the Internet site name and a count field for storing the number of times the Internet site name has been received.

4.   (Currently Amended)   The method of claim 3 further comprising the step of retrieving *r* most frequently used Internet site names according to the value of the count field of each entry.

5.   (Currently Amended)   The method of ~~claim 3~~ claim 1 wherein if the table is full and the Internet site name is not in the table, replace one of the *q* least frequently used entries according to the value of the count field of each entry.

6.   (Currently Amended)   An apparatus for ~~building a table to select~~ caching resources of *r* most frequently used Internet site names ~~in a resource~~ ~~delivery site, so that the resource delivery site can cache the corresponding~~

~~resources from the *r* most frequently used Internet sites in advance,~~ the ~~apparatus~~ comprising:

(a)   a memory for storing [[the]] <u>a</u> table having *n* entries where *n* ≥ *r*, <u>where *r* is the number of most frequently used Internet sites</u> and each entry comprises a name field; and

(b)   a processor for<u>,</u>

<u>receiving an Internet site name</u><u>,</u> ~~from a packet and~~

storing the Internet site name into the name field of an entry in the table,

~~characterized in that if the table is full and the Internet site name is not in the table, the processor replaces one of the *q* least frequently used entries, where *q* < *n*~~

<u>selecting an entry from a set of replaceable entries in the table if the table is full and the Internet site name is not in the table; and</u>

<u>replacing the selected entry with the Internet site name entry.</u>

7.   (Original)   The apparatus of claim 6 wherein the Internet site name is a URL (Uniform Resource Locator).

8.   (Currently Amended)   The apparatus of claim 6 wherein if the table is full and the Internet site name is not in the table, the processor randomly selects one of [[the]] *q* at least frequently used entries for replacement <u>from the set of replaceable entries.</u>

9.   (Currently Amended)   The apparatus of claim 6 wherein if the table is full and the Internet site name is not in the table, the processor replaces the least frequently used entry among [[the]] q least frequently used entries <u>from the set of replaceable entries</u>.

10.   (Original)   The apparatus of claim 6 wherein each entry in the table further comprises a count field for storing the number of times the associated Internet site name in the entry has been received.

11.   (Original)   The apparatus of claim 10 wherein if the Internet site name is in one of the entries, the processor increments the value of the count field.

12.   (Original)   The apparatus of claim 11 wherein the processor sorts the entries in the table into an order according to the value of the count field of each entry.

13.   (Original)   The apparatus of claim 12 wherein the order is descending, whereby the r most frequently used Internet site names are in the first r entries.

14.   (Original)   The apparatus of claim 12 wherein the sorting method is a bubble sort method.

15.   (Original)   The apparatus of claim 10 wherein the processor retrieves the r most frequently used Internet site names from the top r entries according to the value of the count field of each entry.

16.   (Currently Amended)   An apparatus for ~~building a table to select~~ caching resources of r most frequently used Internet site names ~~in a resource delivery site, so that the resource delivery site can cache the corresponding resources from the r most frequently used Internet sites in advance~~, the apparatus comprising:

(a)   a receiver for receiving an Internet site name;

(b)   a processor for converting the Internet site name into a <u>hash</u> number and storing the number into an entry in [[the]] <u>a</u> table; and

(c)   a memory for storing the table having n entries where n ≥ r, <u>where r is the number of most frequently used Internet sites,</u> each entry in the table comprising a number field for the number, a name field for the Internet site name and a count field for counting the number of times the Internet site name is received, ~~characterized in that~~ <u>wherein</u>

<u>(d)   the processor further selects an entry from a set of replaceable entries in the table</u> if the table is full and the number is not in the table~~, the processor replaces one of the q least frequently used entries according to the value of the count field of each entry, where q < n~~ <u>and replaces the selected</u>

entry with the hash number entry according to the value of the count field of each entry.

17. (Original) The apparatus of claim 16 wherein the Internet site name is a URL (Uniform Resource Locator).

18. (Original) The apparatus of claim 16 wherein if the number is in one of the entries, the processor increments the value of the count field.

19. (Original) The apparatus of claim 16 wherein the processor retrieves the r most frequently used Internet site names from the top r entries according to the value of the count field of each entry.

20. (Original) The apparatus of claim 16 wherein the processor sorts the entries in the table into an order according to the value of the count field of each entry.

21. (Cancelled)

22. (Original) The apparatus of claim 16 wherein if the number is not in the table and the table is not full, the processor stores the number and the Internet site name in the respective fields of an empty entry.

23. (Original)   The apparatus of claim 16 wherein if the number is in an entry and the value of the count field of that entry is greater than a threshold, the processor stores the Internet site name in that entry.

24. (Currently Amended)   The apparatus of claim 16 wherein if the table is full and the number is not in the table, the processor randomly selects one of [[the]] q least frequently used entries for replacement from the set of replaceable entries.

25. (Currently Amended)   The apparatus of claim 16 wherein if the table is full and the number is not in the table, the processor replaces the entry with the smallest value of the count field among [[the]] q least frequently used entries from the set of replaceable entries.

26. (Original)   The apparatus of claim 16 wherein the table comprises q sub-tables where $n>q>1$, each sub-table has n/q entries and pointed to by an address ranging from 0 to $q-1$, the number is searched or stored in the sub-table pointed to by the address produced by taking a modulo operation on the number by q, if the sub-table is full and the number is not in the sub-table, the processor replaces one of the bottom m/q entries according to the value of the count field of each entry, and retrieves the r most frequently used Internet site names from the top r entries among the q sub-tables according to the value of the count field of each entry.

27.   (Currently Amended)   A ~~computer program product having a~~ computer readable medium having computer program logic recorded thereon for building a table to select r most frequently used Internet site names, the computer program ~~product~~ logic comprising:

(a)   a computer program code segment for receiving an Internet site name ~~from a packet~~;

(b)   a computer program code segment for converting the received Internet site name into a hash number;

(c)   a computer program code segment for storing the number in the table having n entries where $n \geq r$, each entry in the table comprising a number field for the number, a name field for the received Internet site name and a count field for counting the number of times the Internet site name has been received, ~~characterized in that~~ wherein

(d)   the computer code segment for storing further selects an entry from a set of replaceable entries in the table if the table is full and the number is not in the table~~, the storing computer program code segment replaces one of the q least frequently used entries according to the value of the count field of each entry, where $q < n$~~ and replaces the selected entry with the new entry according to the value of the count field of each entry.

28. (Currently Amended) The computer ~~program product~~ <u>readable medium</u> of claim 27 wherein the received Internet site name is a URL (Uniform Resource Locator).

29. (Currently Amended) The computer ~~program product~~ <u>readable medium</u> of claim 27 wherein if the number is in one of the entries, the storing computer program code segment increments the value of the count field.

30. (Currently Amended) The computer ~~program product~~ <u>readable medium</u> of claim 27 <u>wherein the logic</u> further ~~comprising~~ <u>comprises</u> a computer program code segment for retrieving the r most frequently used Internet site names from the top r entries according to the value of the count field of each entry.

31. (Currently Amended) The computer ~~program product~~ <u>readable medium</u> of claim 27 <u>wherein the logic</u> further ~~comprising~~ <u>comprises</u> a computer program code segment for sorting the entries in the table into an order according to the value of the count field of each entry.

32. (Cancelled)

33.   (Currently Amended)   The computer ~~program product~~ <u>readable</u> <u>medium</u> of claim 27 wherein if the number is in the table and the table is not full, the storing computer program code segment stores the number and the received Internet site name in the respective fields of an empty entry.

34.   (Currently Amended)   The computer ~~program product~~ <u>readable</u> <u>medium</u> of claim 27 wherein if the number is in an entry and the value of the count field in that entry is greater than a threshold, the storing computer program code segment stores the received Internet site name in the name field of that entry.

35.   (Currently Amended)   The computer ~~program product~~ <u>readable</u> <u>medium</u> of claim 27 wherein if the table is full and the number is not in the table, the storing computer program code segment randomly selects one of [[the]] q least frequently used entries for replacement <u>from the set of</u> <u>replaceable entries</u>.

36.   (Currently Amended)   The computer ~~program product~~ <u>readable</u> <u>medium</u> of claim 27 wherein if the table is full and the number is not in the table, the storing computer program code segment replaces the entry with the smallest received count among [[the]] q least frequently used entries.

37.   (New) The method of claim 1 wherein the cached resource is a Hypertext Markup Language (HTML) file.

38.   (New) The method of claim 1 wherein the cached resource is an audio file.

39.   (New) The apparatus of claim 6 wherein the resources include an HTML file.

40.   (New) The apparatus of claim 6 wherein the resources include an audio file.

41.   (New) The apparatus of claim 16 wherein the resources include an HTML file.

42.   (New) The apparatus of claim 16 wherein the resources include an audio file.